A Signature-based Intrusion Detection System for the Internet of Things

Philokypros P. Ioulianou*, Vassilios G. Vassilakis*, Ioannis D. Moscholios†, Michael D. Logothetis‡
*Dept. of Computer Science, University of York, York, United Kingdom
†Dept. of Informatics & Telecommunications, University of Peloponnese, Tripolis, Greece
‡Dept. of Electrical & Computer Engineering, University of Patras, Patras, Greece

***Abstract*—Internet of Things (IoT) is envisioned as a transformative approach with a wide range of applications in various sectors such as home automation, industrial control, and agriculture. It promises innovative business models and improved user experience. However, as evidenced by recent attacks such as the Mirai botnet, IoT networks and systems remain very vulnerable and require stronger protection mechanisms. Furthermore, due to processing, memory, and power constraints of typical IoT devices, traditional Internet security mechanisms are not always feasible or appropriate. In this work, we are concerned with designing an Intrusion Detection System (IDS) for protecting IoT networks from external threats as well as internal compromised devices. Our proposed design adopts a signature-based intrusion detection approach and involves both certralised and distributed IDS modules. Using the Cooja simulator, we have implemented a Denial of Service (DoS) attack scenario on IoT devices. This scenario exploits the RPL protocol, which is widely used for routing in low-power networks, including IoT networks. In particular, we have implemented two variants of DoS attacks, namely "Hello" flooding and version number modification. As shown by simulation results, these attacks may impact the reachability of certain IoT devices and their power consumption.**

I. Introduction

The rapid growth of smart devices led to an Internet-connected world called "Internet of Things" (IoT) [1]. In the literature, these devices are usually referred to as "things", IoT devices, smart devices, sensors, or motes. IoT devices are typically resource-constrained (low processing power, small batteries, limited memory), but can connect to the Internet, exchange data, and perform limited computations. Some known IoT application domains are smart home, industrial control, health monitoring, and smart grid [2, 3]. However, apart from the obvious benefits, IoT brought new security and privacy challenges.

According to a recent report by Gemalto [4], securing smart devices is not a priority for manufacturers. This enables the attackers to target IoT devices with weak security measures. A recent example of a cyber attack is the Mirai botnet [5], which exploited the default passwords in many IoT devices (IP cameras, digital video recorders) and coordinated a Distributed Denial of Service (DDoS) attack to many targets. This and other similar incidents indicate that the security of IoT devices and networks must be re-examined and appropriate solutions should be developed to protect businesses, consumers, and critical infrastructure.

An Intrusion Detection System (IDS) is a security technology that monitors networks or systems for malicious activity or policy violations. In the recent years, IDSes have attracted the attention of security researchers and practitioners for protecting IoT devices [6]. In the literature, three types of intrusion detection methods are typically distinguished, namely signature-based, anomaly-based, and specification-based [7]. Hybrid approaches, that combine two or more methods, are also gaining popularity.

Signature-based IDS may detect an attack/intrusion if the attack's signature is already stored in the internal database. These systems can detect known attacks very accurately and this is the reason why they are widely used in the industry. Anomaly-based detection tries to recognise malicious behaviour. It needs the previous creation of profiles for defining the normal behaviour of users, hosts, or networks. Therefore, the required data is collected and stored in a database during the normal operation. Specification-based detection is similar to anomaly-based detection. In this method, the normal behaviour is defined by taking into account the functionalities and the security policies of the system. A profile with the expected normal behaviour is created and regularly consulted.

In this paper, we propose a new signature-based IDS for the detection of DoS and routing attacks in IoT networks. This IDS follows a hybrid placement strategy for IDS modules. That is, it involves both centralised and distributed components. In particular, the main router runs the detection module and other lightweight modules are deployed in the network in close proximity to the IoT devices for the purposes of traffic monitoring and reporting. One of the advantages of this approach is that no software modification of current devices/sensors is required. Furthermore, all the IDS modules are connected via wired communication channels in order to avoid jamming or other types of wireless attacks.

The rest of the paper is organized as follows: In Section II, we present our considered 7-layer IoT reference model. In Section III, we briefly discuss the attacks that may occur in IoT environments. In Section IV, we present the most important currently available IDS solutions. In section V, we describe our DoS attack implementation in Cooja simulator and present the simulation results. In Section VI, we describe our proposed IDS design, including a high-level architecture and its main components. We conclude and discuss our future work in Section VII.

II. Seven-layer IoT Reference Model

Among different proposed IoT reference models, CISCO's 7-layer model [8] is the most detailed one and has been adopted in this work. The layers of CISCO's model are shown in Fig. 1. In this work, we are mostly concerned with Layers 1-3.

Starting from Level 1, physical devices are the smart devices, which send or receive generated/censored data. Level 2 refers to the connectivity between the devices, within the same network or across different networks. IoT devices should be able to reliably transmit data using existing networks. Level 3 activities include data analysis and transformation. In other words, network packets are processed in that level to be understandable to the higher levels. Level 4 is where data is stored and can be used by applications when needed. These data are abstracted in Level 5. This means that data gathered from different sources can be combined and simplified for use in applications. Level 6 is the "Application" level where information is read from Level 5. Applications vary from analytics to system management and control. The highest level is the "Collaboration and Processes" level where the end users are. Making the IoT system useful requires people to collaborate and use IoT applications and their data.

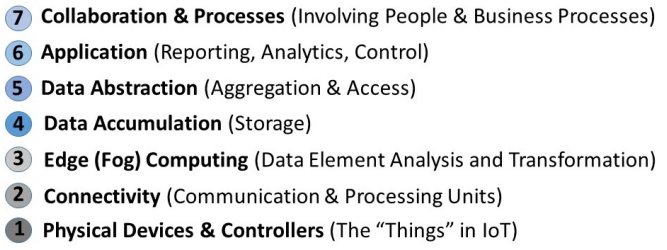

Fig. 1. CISCO's IoT reference model

III. Attacks in IoT

Many IoT devices are still unsecured and attackers could exploit the existing vulnerabilities to cause damage or steal confidential information. In this section, we briefly review the most important types of attacks against IoT devices. For describing attacks, CISCO's 7-layer IoT reference model is assumed.

At the Physical Devices layer, malicious modification of firmware in physical devices could allow the attacker to get access to their data that are stored or in transit. Non-network side-channel attack is another method to exploit the hardware of IoT devices. In that attack, device's electromagnetic signals are monitored by an adversary in order, for example, to reveal the status of the device. Another threat is DoS attacks such as battery draining and resource exhaustion [9]. For example, an adversary may deprive a device from going to sleep by regularly sending "Hello" messages or may exhaust the limited power/memory resources by submitting heavy computation tasks. Last but not least, a node can be cloned by an attacker so that its packets are modified and redirected.

At the Connectivity level, eavesdropping is an attack in which the attacker sniffs network packets and tries to export critical information such as usernames and passwords. Consequently, the attacker can get access to devices, learn about the network infrastructure, or steal crucial data. Routing, replay, and Man-in-the-Middle (MitM) attacks [10] are also dangerous attacks in which the adversary tries to change routing information, and modify, spoof, or drop packets. Furthermore, DoS attacks at the Connectivity level can reduce the performance of the whole IoT network. Specifically, signal jamming and packet flooding are the most common DoS attacks that target device's communication channels. Finally, smart devices can be turned into bots and used for DoS attacks against selected targets, as was the case with the infamous Mirai botnet [5].

At the Edge Computing level, an attacker could inject malicious input to the servers or the network to steal sensitive data. In a similar way, leaking information from a device or server could help the attacker extract information about the types of components and services used in the IoT network. For instance, database errors or warnings reveal important information to the attackers.

All in all, many security issues exist in IoT networks today. Taking into account the well-known Confidentiality, Integrity & Availability (CIA) triad, one of the most important issues to address is to ensure data availability. Data obtained from sensors or other IoT devices should be available when needed. Therefore, DoS and routing attacks should be prevented or eliminated from creating problems to IoT devices and networks. Hence, ensuring data availability and protecting against related attacks is the main focus of this work.

IV. Current IDS Solutions for IoT

IDSes as security measures have been considered by researchers for protecting networks with heterogeneous IoT devices. However, IDSes in traditional networks have different requirements than IoT-based IDSes. Therefore, adapting traditional IDS approaches in IoT environments is not an easy and straightforward task. Features such as limited computation power of smart devices, different network structures, and various developed protocols of IoT devices introduce new challenges that should be addressed by an IoT-based IDS [6]. Below we briefly describe the most important recent IDS solutions for IoT.

Kalis [11] is one of the first developed IDSes that aims at protecting IoT devices irrespective of the IoT protocol or application used. Kalis is a network-based, hybrid signature/anomaly-based, hybrid centralized/distributed, online IDS. The selected detection strategy depends on specific network characteristics. Furthermore, Kalis obtains knowledge from modules installed in the network, and attempts to prevent DoS attacks based on the current network topology, traffic analysis, and mobility information. Kalis can support new protocol standards and allows knowledge sharing between the nodes for better detection. It is implemented on smart routers using the OpenWRT firmware [12]. Evaluation is done using 6 TelosB devices programmed in TinyOS [13]. Experimental

results show that Kalis has better detection performance than traditional IDSes.

Another remarkable work in the field is the SVELTE IDS [14]. This is a signature- and anomaly-based IDS, developed to protect IoT devices from routing attacks based on the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL) [15]. Some of the considered attacks include altering information, sinkhole forwarding, and selective forwarding. SVELTE follows a hybrid module placement approach in which a centralised module, called 6LoWPAN Border Router (6BR), performs heavy calculations and a number of resource-constrained modules are responsible for the monitoring tasks. The 6BR has three components. The first one is the 6LoWPAN Mapper which recreates the network based on the information obtained from IoT nodes. The second component is the IDS one which analyzes information and detects intrusion. The last one is a mini firewall which stops malicious traffic from entering the network. The first and third components are embedded into the IoT nodes.

Despite good progress in developing IoT-based IDSes, current solutions have several limitations. Kalis, for example, requires installation of specialised detection modules for detecting each type of attack. This could create a complex network and could lead in poor detection performance. Moreover, it uses WiFi as communication technology. This means that interference between the smart sensors and Kalis nodes is possible if they are in close proximity. SVELTE has also some limitations as it requires the modification of sensors' software. This, however, would be very inconvenient for networks with large numbers of sensors, which is a typical case in many IoT application domains. All in all, a new technologically improved solution is needed to protect IoT networks from a wide range of possible attacks. The aforementioned limitations have been taken into account when designing our proposed IDS solution.

V. Implementing IoT Attacks in Cooja

In order to design an effective IDS, the first step is to implement a number of attacks and observe their impact on the individual devices and on the network as a whole. After that, by launching attacks with different configuration parameters and intensities, various detection techniques can be implemented, tested, and improved.

For testing and experimentations we use the Cooja simulator [16], which is gaining popularity among IoT researchers. Cooja is particularly suitable for real-world experiments, since the developed applications can be uploaded directly to real hardware. In particular, Cooja can be used to simulate the behaviour of Contiki OS [17] - a popular open source operating system for IoT.

In this work, we have implemented in Cooja two IoT-specific DoS attacks, namely "Hello" flooding and version number modification. These attacks are based on the RPL routing protocol and affect the availability of the network. Cooja already provides an implementation of RPL, called ContikiRPL. RPL organizes routers along a Destination Oriented

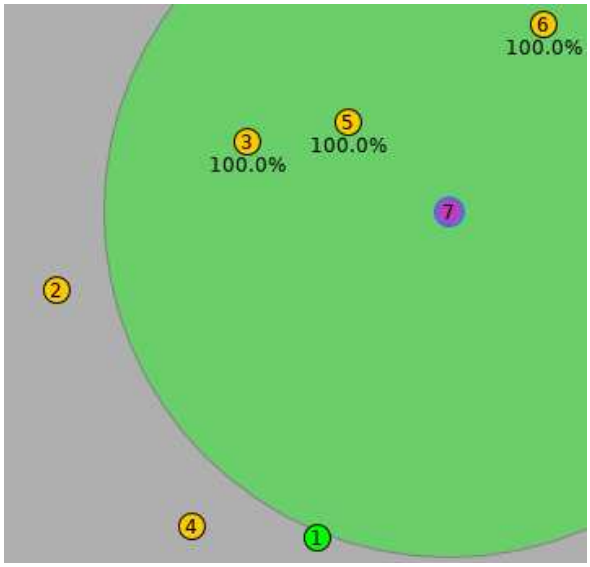

Fig. 2. The network used in Cooja simulations

Directed Acyclic Graph (DODAG) [18]. The graph root initiates the graph formation by periodically originating DODAG Information Object (DIO) messages which it advertises via link-local multicast. DIO messages carry information such as root's identity, routing metrics in use, as well as the originating router's depth (called "rank").

The "Hello" flooding attack in RPL may be launched when a malicious RPL node creates massive amount of traffic by sending DODAG Information Solicitation (DIS) messages to other RPL nodes, causing the recipient nodes to respond by sending DIO messages. As a result, congestion is created in the network and nodes are energy exhausted. Similarly, in version number modification attack the malicious node increases the DODAG version number before forwarding the received DIO messages to the next hop. This causes again resource exhaustion.

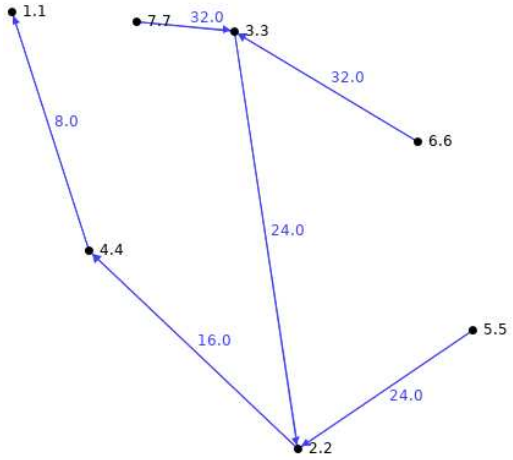

Fig. 3. Scenario 1 (normal operation): Network topology

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	99.91%	0.01%	0.06%
Sky 2	99.79%	0.05%	0.11%
Sky 3	99.92%	0.03%	0.10%
Sky 4	99.86%	0.06%	0.06%
Sky 5	99.91%	0.02%	0.11%
Sky 6	99.79%	0.02%	0.07%
Sky 7	99.88%	0.02%	0.07%
AVERAGE	99.87%	0.03%	0.08%

Fig. 4. Scenario 1 (normal operation): Power consumption measurements

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	99.88%	0.04%	0.12%
Sky 2	99.89%	0.11%	0.36%
Sky 3	99.80%	0.12%	2.12%
Sky 4	99.93%	0.14%	0.14%
Sky 5	99.80%	0.15%	2.09%
Sky 6	99.83%	0.09%	2.04%
Sky 7	99.91%	1.82%	0.32%
AVERAGE	99.86%	0.35%	1.03%

Fig. 6. Scenario 2 (attack): Power consumption measurements

Below we demonstrate two scenarios, simulated in Cooja, showing the effects of the aforementioned DoS attacks. The application used in IoT nodes/sensors is based on the UDP client-server model. Seven Tmote Sky nodes [19] running Contiki OS have been simulated. The network, depicted in Fig. 2, consists of six client nodes with identities (IDs) from 2 to 7 and one server/root node with ID 1. In the first scenario, we do not consider compromised nodes. Each node regularly sends messages to the root node. These messages contain various information about the sending node, such as its temperature and battery indicator. In the second scenario, node 7 is malicious/compromised and performs DoS attacks. In particular, node 7 has been modified to send a large number of DIS messages to its neighbours. Also, it increases the DODAG version number so that the so-called *global repairs* are initiated. This causes the IoT nodes to perform unnecessary computations and consume energy. The simulation time in our experiments in each scenario is 10 minutes.

In the first scenario, the network topology is formed as shown in Fig. 3. The numbers shown on each link indicate the Expected Transmission Count (ETX) value which is the number of transmissions that a node expects to make to a destination in order to successfully deliver a packet. For example, node 4 which is next to the root (node 1) has the ETX value of 8. In Fig. 3 we also observe that messages from node 7 have to be delivered through nodes 3, 2 and 4 in order to reach the root node. Note that node 7 in this scenario is not malicious and is running the same code as all other nodes. Figure 4 shows the power consumption of each node. Measurements were obtained using the PowerTracker tool available in Cooja. As expected, all nodes are almost always on (average 99.87% of the time) and have very low Radio TX and Radio RX values. This is normal for networks of small sizes.

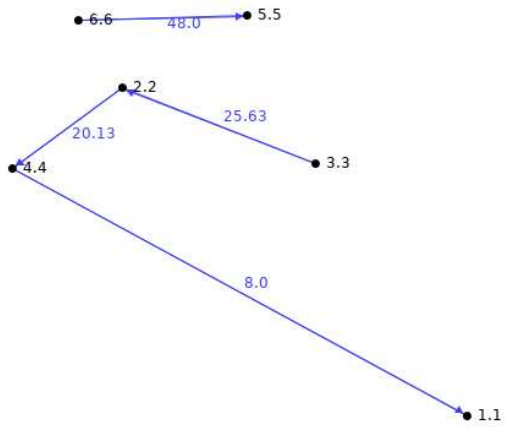

Fig. 5. Scenario 2 (attack): Network topology

In the second scenario, node 7 has been configured to send 80 DIS messages as well as to increase the DODAG version number before forwarding the received DIO messages to the next hop towards the root node. The modification of the version number results in the global repair and the creation of two different DODAGs. Global repair is triggered every few minutes. As a result, routes change rapidly. Hence, the network topology is not fixed and some nodes may be disconnected from the root or other nodes. One such situation is shown in Fig. 5, where nodes 5 and 6 do not have a route to the root in that particular moment. The impact of the attack is shown in Fig. 6, which shows the power consumption measurements. The attack has caused high Radio TX for node 7 and high Radio RX in neighbouring nodes 3, 5, and 6. As a result, both malicious/compromised and neighbouring nodes are energy exhausted.

VI. Proposed Signature-based IDS

A. IDS Architecture and Components

In this section, our proposed IDS solution is described. The IDS is signature-based, because such approaches are more accurate in detecting known attacks, compared to anomaly-based approaches, and typically no heavy computations are required [7].

In addition to the typical sensor nodes, we consider two new types of devices: i) IDS routers for running both the detection module and a firewall, and ii) sensor-like devices, called IDS detectors, for monitoring and sending suspicious traffic to the router. In a typical scenario of a small IoT network, there will be one IDS router and several IDS detectors. The IDS router may also play the role of the Border Router (BR) of the network, as shown in Fig. 7. This means that, sensors requiring to communicate with a server, will send all the requests through the IDS router. All passing traffic is checked by the router who will take the decision whether the sending node is malicious or not.

IDS detectors monitor sensors' traffic to help in detecting malicious nodes. Compromised devices may attempt to inter-

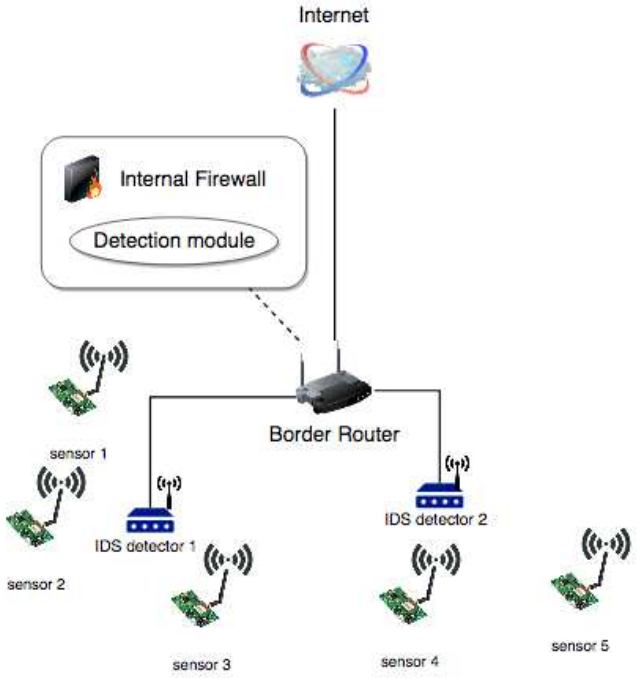

Fig. 7. High-level IDS architecture

rupt the network internally without having to communicate with the BR or external networks. For such cases, IDS detectors will log network traffic and if a node's behaviour resembles a known attack, the related information will be forwarded to the BR for decision making.

In the example of Fig. 7, we have five Tmote Sky sensors and the IDS consisting of one BR and two detectors. The BR is connected to the Internet and includes two components: a firewall and a detection module. These two components help in protecting the network both internally and externally. The detection module runs algorithms to help in decision making, while the firewall creates and enforces rules for blocking malicious sensor requests. The detectors are wired connected to the BR to avoid jamming or eavesdropping via a wireless channel. In cases where a wireless channel between the BR and the detectors is unavoidable or preferable, appropriate secure wireless communication scheme will be in place (e.g., [20]). Any traffic exchanged between the sensors is captured by the nearest detector. Afterwards, a lightweight algorithm is executed to decide if traffic should be forwarded to the BR or not. We assume that detectors will be resource-constrained. Hence, algorithms that require heavy computations or large amounts of memory and storage, would not be suitable.

The combination of BR and detectors helps in capturing traffic from both internal and external communications. For example, some compromised devices may try to communicate with a remote server in order to download commands. Other compromised devices may exchange traffic locally. Our design considers all types of communications so that malicious nodes can be blocked. The BR captures traffic from both WiFi and IEEE 802.15.4 channels. The BR is also able to detect attacks from Zigbee/6LoWPAN devices. Similarly to other signature-based solutions, the proposed IDS stores malicious patterns in the detection module of the BR who is the bridge between the internal network and the Internet. For this reason, it is assumed to have enough computational power to run algorithms for detecting different types of attacks.

B. Attack Mitigation

As mentioned earlier, the proposed IDS aims at detecting and preventing a wide range of different types of attacks. For example, DoS attacks that may occur inside IoT networks to achieve resource exhaustion of the sensor nodes. In addition to that, routing attacks are usually exploiting the RPL protocol which is currently used by smart sensors in many IoT networks. Sinkhole attacks, selective forwarding, and clone ID are some of the widely known routing attacks.

The above mentioned attacks can be mitigated by measuring the Received Signal Strength (RSS), packet data drop rate, or packet sending rate, and by monitoring the number of node IDs in the network [21]. According to reports, these attacks are the ones most commonly used and may affect the availability as well as the integrity of IoT systems. Designing and developing an efficient IDS to protect against DoS and routing attacks in IoT networks is currently an open problem.

As far as the scalability of the proposed IDS is concerned, even in large networks good efficiency is expected. To ensure that, IDS detectors will forward to BR only the necessary traffic. That is, detectors will perform certain calculations (e.g., RSS and packet drop rate) and only if the metric of interest is above a threshold, node's traffic will be forwarded to the BR for further investigation (e.g., signature matching).

C. Detection Module and Firewall

As discussed before, a very important part of the proposed IDS is the detection module within the BR. This module is responsible for classifying a node as malicious or not. The decision is based on the individual information collected for each node. For instance, if a node sends to other nodes too many packets with high rate or the node's signal power is above a threshold, then this node may be considered as malicious. In that case, the node may be removed from the network, its IP will be blacklisted, an appropriate firewall rule will be created, and the network administrator will be alerted. On the other hand, attacks such as selective forwarding are difficult to detect and require more time to identify the malicious node. Signatures of current IoT malware will be stored in the detection module. If a packet matches a known malicious signature or pattern, the destination and source node will be immediate blacklisted.

The firewall inside the BR serves as an additional layer of protection. The firewall will contain rules for blocking IP addresses of nodes which are malicious. Nodes are blocked only if the detection module has information of malicious behaviour. In that case, a new rule with the node's IP is created and the node cannot send or receive data from the Internet.

As far as the placement strategy of IDS modules is concerned, a hybrid approach has been adopted. The centralized node (i.e, BR), stores signatures, analyzes traffic and detects attacks originating from the sensors or coming from the Internet. The decentralized nodes (i.e., IDS detectors), perform lightweight tasks such as monitoring and reporting network data to the BR. This placement strategy helps in capturing traffic and detecting attacks from all network segments. Furthermore, deploying detectors in close proximity to the sensors aims at detecting attack attempts faster and more efficiently rather than waiting the attack traffic to pass via the BR.

VII. Conclusion and Future work

In this paper, we propose a signature-based IDS for IoT networks. We presented a high-level IDS architecture and its main components. The proposed approach involves both centralised and distributed modules for detecting intrusions originating from external networks as well as from internal compromised nodes. The chosen platform for developing and testing the IDS solution is the Cooja simulator, which supports application development for Contiki OS. We have also demonstrated an attack scenario in Cooja, where a compromised node performs DoS attacks that rely on "Hello" flooding and version number modification. As shown, the attack may constitute some nodes unreachable and may negatively impact their power consumption. In our future work we plan to implement and test the proposed design in Cooja. We will also evaluate and improve the IDS performance by reducing the false positives during the attack detection process. Finally, we will import the IDS modules to Contiki OS in order to test its performance in a real-world IoT environment.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[2] V. G. Vassilakis, I. D. Moscholios, J. S. Vardakas, and M. D. Logothetis, "On the digital certificate management in advanced metering infrastructure networks," in *Proc. IEICE Information and Communication Technology Forum (ICTF)*, Poznan, Poland, July 2017, pp. 1–5.

[3] B. A. Alohali and V. G. Vassilakis, "Secure and energy-efficient multicast routing in smart grids," in *Proc. 10th IEEE Int. Conf. on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Singapore, April 2015, pp. 1–6.

[4] Gemalto, *The State of IoT Security*, 2018. [Online]. Available: http://www2.gemalto.com/iot/index.html

[5] "Mirai: What you need to know about the botnet behind recent major DDoS attacks," Symantec Security Response, Oct. 2016.

[6] B. B. Zarpelão, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, "A survey of intrusion detection in Internet of Things," *Journal of Network and Computer Applications*, vol. 84, pp. 25–37, April 2017.

[7] H.-J. Liao *et al.*, "Intrusion detection system: A comprehensive review," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16–24, Jan. 2013.

[8] CISCO, "The Internet of Things reference model," *White Paper*, 2014.

[9] Y. Yang *et al.*, "A survey on security and privacy issues in Internet-of-Things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, Oct. 2017.

[10] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of Things security: A survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, June 2017.

[11] D. Midi, A. Rullo, A. Mudgerikar, and E. Bertino, "Kalis - A system for knowledge-driven adaptable intrusion detection for the Internet of Things," in *Proc. IEEE 37th Int. Conf. on Distributed Computing Systems (ICDCS)*, Atlanta, GA, USA, June 2017, pp. 656–666.

[12] "OpenWRT: A Linux OS for Embedded Devices." [Online]. Available: https://openwrt.org/

[13] "TinyOS: An OS for Embedded, Wireless Devices." [Online]. Available: https://github.com/tinyos/tinyos-main

[14] S. Raza, L. Wallgren, and T. Voigt, "SVELTE: Real-time intrusion detection in the Internet of Things," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2661–2674, June 2013.

[15] T. Winter *et al.*, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," Internet Requests for Comments, RFC 6550, March 2012. [Online]. Available: http://www.rfc-editor.org/rfc/rfc6550.txt

[16] F. Osterlind *et al.*, "Cross-level sensor network simulation with Cooja," in *Proc. 31st IEEE Int. Conf. on Local Computer Networks*, Tampa, FL, USA, Nov. 2006, pp. 641–648.

[17] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki - A lightweight and flexible operating system for tiny networked sensors," in *Proc. 29th IEEE Int. Conf. on Local Computer Networks*, Tampa, FL, USA, Nov. 2004, pp. 455–462.

[18] E. Baccelli, M. Philipp, and M. Goyal, "The P2P-RPL routing protocol for IPv6 sensor networks: Testbed experiments," in *Proc. 19th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, Croatia, Sept. 2011, pp. 656–666.

[19] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling ultra-low power wireless research," in *Proc. 4th Int. Symp. on Information Processing in Sensor Networks*, Boise, ID, USA, April 2005, pp. 364–369.

[20] B. A. Alohali and V. G. Vassilakis, "A secure scheme for group communication of wireless iot devices," in *Proc. 11th IEEE/IET Int. Symp. on Communication Systems, Networks, and Digital Signal Processing (CSNDSP)*, Budapest, Hungary, July 2018, pp. 1–6.

[21] A. Rghioui, A. Khannous, and M. Bouhorma, "Denial-of-Service attacks on 6LoWPAN-RPL networks: Threats and an intrusion detection system proposition," *Journal of Advanced Computer Science & Technology*, vol. 3, no. 2, pp. 143–153, 2014.